

**PROSES AUTOMASI ETL DATA CORE PADA PROYEK
TECHNICAL SUPPORT ASSISTANT NETWORK DATA
MART**

Laporan Kerja Praktik

Disusun Untuk Memenuhi Salah Satu Tugas Mata Kuliah Kerja Praktik

Pendidikan Diploma Program Studi D3 Teknik Informatika

Jurusan Teknik Komputer dan Informatika



Disusun oleh:

Muhamad Rafli Nur Ikhsan NIM 201511048

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA
PROGRAM STUDI DIPLOMA III TEKNIK INFORMATIKA
POLITEKNIK NEGERI BANDUNG
TAHUN 2022**

KATA PENGANTAR

Segala puji dan syukur kehadiran Allah Swt. Yang telah memberi kesehatan dan kemudahan untuk penulis dalam menyusun laporan Kerja Praktik (KP) kali ini. Selesaiannya laporan KP ini tidak terlepas dari bantuan banyak pihak yang telah memberikan masukan kepada penulis. Maka dari itu, penulis mengucapkan terimakasih kepada:

1. Bapak Achmad Sobar Sururi selaku Direktur PT Neural Technologies Indonesia yang telah memberi kesempatan kepada penulis untuk melaksanakan KP di perusahaan PT Neural Technologies Indonesia;
2. Bapak Usep Ependi selaku *Project Manager* pada saat KP di PT Neural Technologies Indonesia;
3. Kang Kemal Nugraha selaku pembimbing industri pada saat KP di PT Neural Technologies Indonesia;
4. Bapak Akhmad Bakhrun S.Kom., M.T., Ibu Ida Suhartini, S.Kom., MMSI, selaku koordinator KP D3 JTK tahun 2022, yang telah bekerja keras dalam rangkaian kegiatan KP ini, dimulai dari persiapan KP hingga penyelesaian KP tahun 2022;
5. Bapak Akhmad Bakhrun, S.Kom, M.T., selaku pembimbing jurusan, yang telah membimbing, mengarahkan, memberikan solusi selama pelaksanaan KP serta penyusunan laporan KP ini;
6. Keluarga penulis yang telah memberikan dukungan selama KP;
7. Teman-teman yang sudah membantu penulis dalam pelaksanaan KP sekaligus penyusunan laporan KP ini.

Penulis menyadari bahwa laporan ini masih memiliki banyak kekurangan, baik dari isi materi maupun penyajiannya. Maka dari itu, kritik dan saran sangat penulis terima untuk menyempurnakan laporan ini. Semoga laporan ini dapat memberikan manfaat bagi penulis maupun yang membacanya.

Bandung, 16 November 2022

Muhamad Rafli Nur Ikhsan

RINGKASAN

Technical Support Assistant Network Data Mart Telkomsel adalah sebuah proyek untuk mengelola dan melakukan monitoring terhadap data Telkomsel, salah satunya adalah Data Core. Data Core merupakan *source* data utama yang mana berupa *database* yang menyimpan jumlah *subscriber* di suatu area, jumlah *payload*, jumlah SMS, dll. Setiap harinya data dari Data Core dipindahkan ke *table destination* untuk dilaporkan kepada klien kemajuan dari produknya, pemindahan dilakukan secara manual sehingga membutuhkan waktu yang cukup lama hingga semua data berhasil dipindahkan. Automasi *Extract, Transform, and Load* (ETL) diusulkan guna mempersingkat waktu pemindahan untuk memudahkan dan meningkatkan produktivitas pekerja.

Keyword: Data Core, automasi, ETL.

DAFTAR ISI

KATA PENGANTAR.....	i
RINGKASAN	iii
DAFTAR GAMBAR.....	vi
BAB I.....	1
I.1 Latar Belakang	1
I.2 Perumusan Masalah.....	2
I.3 Ruang Lingkup	2
I.4 Tujuan.....	2
I.5 Tahap Pengerjaan Tugas	2
I.6 Lokasi dan Waktu Pelaksanaan	2
I.7 Sistematika Penulisan.....	2
BAB II	4
II.1 Tujuan Umum Perusahaan	4
II.2 Struktur Organisasi Perusahaan.....	5
II.3 Deskripsi Kerja.....	5
II.3.1 Jam Kerja	5
II.3.2 Lingkungan Kerja.....	5
II.3.3 Budaya Kerja.....	6
II.3.3.1 <i>Standup Meeting</i>	6
II.3.3.2 Makan	6
II.3.3.3 Halal Bihalal	7
II.3.3.4 17 Agustusan.....	7
II.3.3.5 <i>Mini Soccer</i>	8
II.3.3.6 Bulu Tangkis.....	8
BAB III.....	10
III.1 Konsep Dasar	10
III.1.1 Python	10
III.1.2 Extract Transform Load (ETL)	11
III.2 Tools yang Digunakan.....	12

III.2.1	Visual Studio Code	12
III.2.2	Telegram	12
III.2.1	Task Scheduler	13
III.2.3	Dbeaver	14
BAB IV	15
IV.1	Analisis Sistem Berjalan	15
IV.2	Evaluasi Hasil Analisis.....	16
IV.3	Kesimpulan Kebutuhan Aplikasi yang Dikembangkan	16
BAB V	17
V.1	Menentukan <i>Source Table</i>	17
V.2	Menentukan Pengurangan dan Penambahan Kolom.....	17
V.1	Menentukan Jadwal Pengambilan Data	17
BAB VI	18
VI.1	Proses Pengambilan data	18
VI.2	Proses Pengelolaan Data	19
VI.3	Proses <i>Ingest</i> Data ke <i>Database</i>	20
VI.4	Proses Automasi Run File	21
BAB VII	24
VII.1	Kesimpulan	24
VII.2	Saran	24
DAFTAR PUSTAKA	25

DAFTAR GAMBAR

Gambar 1. Bagan Perusahaan	5
Gambar 2. Standup Meeting	6
Gambar 3. Tempat Makan.....	7
Gambar 4. Halal Bihalal.....	7
Gambar 5. 17 Agustus.....	8
Gambar 6. Mini Soccer	8
Gambar 7. Bulu Tangkis	9
Gambar 8. Logo Python	10
Gambar 9. Logo Visual Studio Code	12
Gambar 10. Logo Telegram	13
Gambar 11. Logo task scheduler.....	13
Gambar 12. Logo Dbeaver	14
Gambar 13. Codingan 1	18
Gambar 14. Codingan 2	18
Gambar 15. Output file 1	19
Gambar 16. Codingan 3	19
Gambar 17. Output file 2	20
Gambar 18. Codingan 3	20
Gambar 19. File .bat.....	21
Gambar 20. Task scheduler 1	22
Gambar 21. Task scheduler 2.....	22
Gambar 22. Task scheduler 3.....	23

BAB I

PENDAHULUAN

I.1 Latar Belakang

PT. Neural Technologies Indonesia mempunyai proyek bernama *Technical Support Assistant Network Data Mart* Telkomsel, proyek tersebut adalah sebuah proyek untuk mengelola dan melakukan monitoring terhadap data Telkomsel, salah satunya adalah Data Core. Data Core merupakan *source data* utama yang mana berupa *database* yang menyimpan jumlah *subscriber* di suatu area, jumlah *payload*, jumlah *Short Message Service* (SMS), dll. Data ini pun setiap harinya diambil dan dipindahkan ke *destination table* untuk dilaporkan kepada *client* agar *client* dapat mengetahui kemajuan jaringan setiap harinya.

Saat ini pengambilan data untuk dilaporkan kepada *client* masih diambil secara manual. Data diambil dari *database core* lalu diolah sesuai dengan keperluan informasi dan pada akhirnya disimpan di *table destination*. Pengambilan data ini dilakukan secara manual satu persatu dan dilakukan rutin setiap harinya agar klien lebih mudah untuk mengetahui kemajuan produk setiap harinya..

Pengolahan data seperti ini tentunya akan memakan waktu yang sangat lama. Misalnya satu *table* memiliki 100000 data, jika dilakukan pemindahan data secara manual mungkin akan memakan waktu minimalnya 1 jam, terlebih lagi pemindahan data ini baru dari satu *table*, dan pemindahan ini dilakukan rutin setiap hari. Jika terus seperti ini maka akan banyak sekali waktu yang terbuang hanya untuk pemindahan data ini.

Automasi *Extract, Transform, and Load* (ETL) diajukan untuk mempersingkat proses pemindahan Data Core ini. Dengan dilakukannya automasi ETL diharapkan proses pemindahan menjadi singkat dan dapat dilakukan secara otomatis oleh sistem yang ada.

I.2 Perumusan Masalah

Proyek Technical Support Assistant Network Data Mart ini mengelola data yang sangat banyak, sehingga timbul masalah “Bagaimana cara menerapkan proses automasi ETL pada pengelolaan Data Core?”.

I.3 Ruang Lingkup

Ruang lingkup pembuatan automasi ETL ini adalah:

1. Automasi ETL yang dibuat untuk Data Core.
2. Automasi ETL yang dibuat sampai data dari *source* berhasil pindah ke *destination*.

I.4 Tujuan

Tujuan utama dari proyek ini adalah membuat proses automasi ETL pada pengelolaan Data Core.

I.5 Tahap Pengerjaan Tugas

Untuk menyelesaikan tugas yang diberikan ketika pelaksanaan Kerja Praktek (KP) diperlukan eksplorasi terlebih dahulu dikarenakan sebelumnya belum pernah melakukan hal yang serupa. Lalu dilakukan terlebih dahulu analisis dan perancangan sebelum masuk ke tahap implementasi.

I.6 Lokasi dan Waktu Pelaksanaan

Lama pelaksanaan Kerja Praktik ini adalah 10 minggu, terhitung dari tanggal 4 Juli hingga 9 September. Untuk pelaksanaan Kerja Praktik ini terbagi menjadi dua, yang pertama *Work From Home* selama 7 minggu, dan *Work From Office* di 3 minggu terakhir pelaksanaan Kerja Praktik di kantor PT. Neural Technologies Indonesia tepatnya di Tebet, Jakarta Selatan.

I.7 Sistematika Penulisan

Sistematika penulisan dibuat untuk memudahkan penulisan laporan KP, berikut sistematika pembahasan dalam penelitian ini:

BAB I PENDAHULUAN

Pada bab ini, dijelaskan mengenai latar belakang, perumusan masalah, ruang lingkup, tujuan, dan tahapan dari pengerjaan proyek yang dikerjakan. Kemudian terdapat dokumentasi terkait lokasi, waktu KP dan sistematika penulisan laporan KP.

BAB II PROFIL PERUSAHAAN

Pada bab ini, dijelaskan profil perusahaan PT Neural Technologies Indonesia. Mulai dari tujuan umum perusahaan, struktur organisasi, dan deskripsi kerja. Pada deskripsi kerja terdapat penjelasan kembali terkait jam kerja, lingkungan kerja, dan budaya kerja.

BAB III PENGETAHUAN DAN PERANGKAT PENDUKUNG

Pada bab ini, dijelaskan konsep dasar dan tools yang digunakan pada saat pengembangan. Mulai dari *tools* untuk pengembangannya, hingga teknologi yang menunjang pada saat pengembangan proyek.

BAB IV ANALISIS DAN EVALUASI

Pada bab ini, dijelaskan mengenai analisis dari proses ETL secara umum. Analisis tersebut terdiri dari analisis sistem berjalan, evaluasi hasil analisis, dan yang terakhir kesimpulan kebutuhan aplikasi yang dikembangkan.

BAB V PERANCANGAN

Pada bab ini, dijelaskan mengenai perancangan ETL yang akan diterapkan pada pengelolaan Data Core. Bab ini menjelaskan cara untuk menentukan *source table*, menentukan pengudaran dan penambahan kolom, dan menentukan penjadwalan *run file* otomatis.

BAB VI IMPLEMENTASI

Pada bab ini, dijelaskan mengenai implementasi dari analisis dan perancangan sebelumnya pada pengelolaan Data Core. Implementasi di sini dibagi menjadi beberapa bagian, yaitu proses pengambilan data, proses pengelolaan data, proses *ingest* data ke *database*, dan yang terakhir proses automasi *run file*.

BAB VII PENUTUP

Bab ini adalah bab terakhir pada laporan KP ini. Bab ini terdiri dari kesimpulan dan saran. Isi dari kesimpulan adalah menjelaskan apakah masalah

sudah selesai atau belum, dan isi dari saran adalah usulan terkait pengerjaan proyek kedepannya.

BAB II

PROFIL PERUSAHAAN

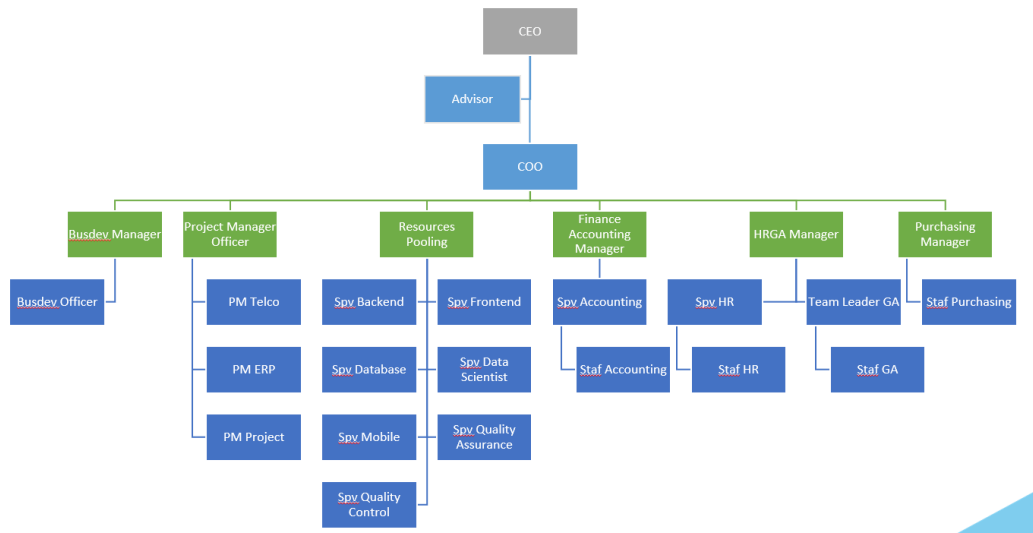
II.1 Tujuan Umum Perusahaan

PT Neural Technologies Indonesia (NTI) merupakan perusahaan berbasis IT yang bergerak di bidang multi-disiplin mulai dari Telekomunikasi, Managed Service / Man-Power Supply, Infrastruktur & General Supply, dan Alat Kesehatan. Selain itu, PT NTI menawarkan produk *Information and Technology* (IT) seperti *Enterprise Resource Planning* (ERP) *software*, *software* analitik sistem, teknologi sistem informasi, dan layanan pengelolaan sistem. PT NTI berdiri sejak tahun 2007 dan berlokasi di Gedung SOHO Pancoran, Kecamatan Tebet Jakarta Selatan.

PT NTI memiliki slogan “NTI, Your IT Solution” sehingga selalu berusaha memberikan yang terbaik terhadap klien agar menjadi solusi IT bagi segala pihak yang ada. Selain selalu berusaha memberikan yang terbaik terhadap klien, PT NTI pun sangat peduli terhadap karyawannya. Sesuai yang diucapkan oleh direktur utama yaitu ingin menjadikan NTI sebagai rumah kedua bagi karyawannya, yang berarti selalu berusaha pula membuat karyawannya nyaman agar mempunyai semangat dan motivasi yang kuat saat bekerja.

II.2 Struktur Organisasi Perusahaan

Adapun struktur organisasi di PT Neural Technologies Indonesia adalah sebagai berikut:



Gambar 1. Bagan Perusahaan

Untuk membangun sebuah sistem yang solid agar menjadi solusi IT bagi segala pihak, PT NTI memiliki 3 divisi utama yaitu Telco, ERP dan Proyek lainnya. Khusus untuk proyek yang dikerjakan untuk laporan ini ada di bawah naungan divisi Telco.

II.3 Deskripsi Kerja

PT NTI memiliki kondisi kerja atau *working condition* sebagai berikut.

II.3.1 Jam Kerja

Untuk jam kerja di PT NTI dimulai pukul 09.00 WIB hingga 18.00 WIB dengan adanya istirahat selama 1 jam pada 12.00 WIB hingga 13.00 WIB.

II.3.2 Lingkungan Kerja

Untuk lingkungan kerja di PT NTI sangat nyaman dan asri, bangunannyapun cukup luas yang terdiri dari beberapa ruangan yang menunjang untuk bekerja secara leluasa. Fasilitas kantor pun lengkap, mulai dari *wifi*, listrik, kursi yang nyaman, *smart tv*, mesin pembuat kopi, apartemen, makan siang dan sore dan transportasi berupa motor.

II.3.3 Budaya Kerja

II.3.3.1 *Standup Meeting*

Standup Meeting merupakan sebuah pertemuan rutin yang dilakukan pagi hari guna mengetahui progress masing-masing individu dalam sebuah proyek.

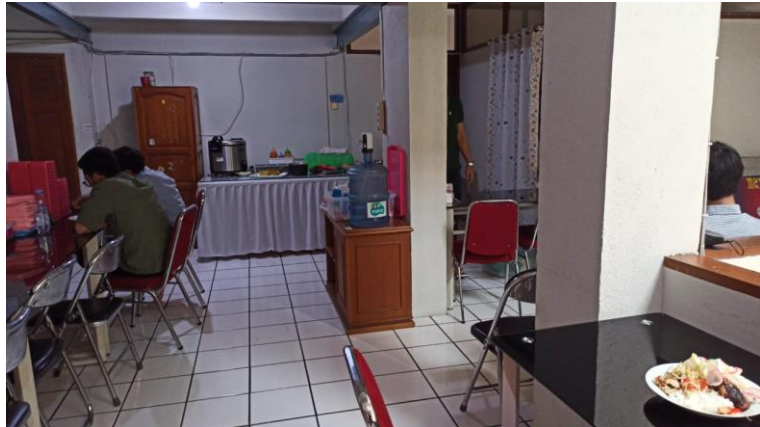
Poin dari *Standup Meeting* ini adalah memberitahu apa yang akan dikerjakan di hari ini, apa yang sedang atau masih dikerjakan dari hari kemarin dan memberi tahu apa yang sudah selesai dikerjakan.



Gambar 2. Standup Meeting

II.3.3.2 **Makan**

Di PT. NTI memiliki budaya untuk makan bersama di siang hari pada saat istirahat, dan di sore hari sebelum pulang kerja. Makan bersama ini berupa prasmanan dimana setiap pegawai mengambil nasi dan lauknya masing-masing.



Gambar 3. Tempat Makan

II.3.3.3 Halal Bihalal

Halal Bihalal merupakan acara saling memaafkan yang diselenggarakan ketika awal masuk kerja ketika selesai libur Idul Fitri. Acara ini rutin diadakan setiap setahun sekali.



Gambar 4. Halal Bihalal

II.3.3.4 17 Agustusan

PT NTI selalu rutin merayakan hari 17 Agustus guna membangkitkan semangat kemerdekaan Negara Kesatuan Republik Indonesia. Perayaan tersebut diisi dengan mengadakan lomba-lomba yang melatih kerja sama individu dan wawasan kebangsaan.



Gambar 5. 17 Agustus

II.3.3.5 Mini Soccer

Salah satu olahraga rutin yang ada di PT. NTI adalah *Mini Soccer*, kegiatan ini dilaksanakan setiap seminggu sekali tepatnya di Senin malam. *Mini Soccer* ini boleh diikuti oleh seluruh pegawai pria yang ingin mengikuti *Mini Soccer* ini.



Gambar 6. Mini Soccer

II.3.3.6 Bulu Tangkis

Kegiatan olahraga terakhir yang diadakan rutin seminggu sekali adalah bulu tangkis. Berbeda dengan *Mini Soccer*, bulu

tangkis bisa diikuti oleh wanita, hanya saja harinya yang berbeda. Untuk pegawai wanita dilakukan rutin setiap Rabu malam, dan untuk pegawai pria dilakukan rutin setiap Kamis malam.



Gambar 7. Bulu Tangkis

BAB III

PENGETAHUAN DAN PERANGKAT PENDUKUNG

III.1 Konsep Dasar

III.1.1 Python

Python adalah bahasa pemrograman serbaguna yang bisa dijalankan pada hampir semua arsitektur sistem, dan bisa digunakan untuk berbagai aplikasi di banyak bidang, mulai dari *web development* hingga *machine learning*. Selain keserbagunaannya, Python adalah bahasa pemrograman yang cukup mudah dipelajari oleh para pemula sehingga menjadi salah satu bahasa pemrograman yang paling populer.



Gambar 8. Logo Python

Sumber : <https://codepolitan.com/blog/5-library-python-untuk-data-science-59b774b6cad97>

Pada Kerja Praktik ini, python digunakan sebagai bahasa pemrograman utama untuk melakukan automasi ETL. Banyak sekali *library* yang terdapat pada python ini, namun untuk melakukan pengembangan proyek ini hanya membutuhkan *library* Pandas. Pandas adalah sebuah *library* yang digunakan untuk memproses data yang meliputi pembersihan data, manipulasi data, hingga melakukan analisis data.

III.1.2 Extract Transform Load (ETL)

ETL adalah proses pengambilan data dari berbagai sumber untuk dipindahkan ke satu tujuan yang sama. Dalam proses ETL ini memiliki 3 tahapan, yang pertama adalah ekstraksi data untuk mengumpulkan data dari sumber, yang kedua adalah transformasi untuk mengelola data sesuai yang diinginkan, dan yang terakhir adalah memuat data untuk memindahkan data yang sudah diolah ke tujuan yang baru.

Salah satu konsep ETL yang umum dilakukan adalah mengelola data dari *database* ke *database* lagi, yang artinya data yang menjadi *source* berasal dari *database* dan yang menjadi *destination* merupakan *database* lagi. Pada proses ekstraksi, pengambilan data berasal dari *database* yang dipilih menjadi *source*, data tersebut diambil dan diubah ke bentuk *dataframe*.

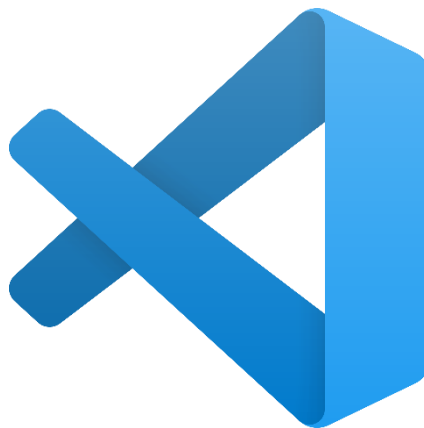
Ketika data sudah berhasil terpindahkan ke *dataframe* lalu dilakukannya proses transformasi data untuk merapihkan data sebelum dimuat ke *table destination*. Proses transformasi tersebut meliputi pembersihan data, standarisasi data sesuai dengan aturan yang sudah ditentukan, dedplikasi untuk melakukan pengecekan terhadap data yang sama apakah akan dibuang atau tidak, dan aturan tambahan lainnya untuk meningkatkan kualitas data.

Yang terakhir adalah proses memuat data. Pada proses ini data yang sudah di transformasi dimuat ke *table destination* yang sudah ditentukan, data tersebut dapat dimuat sekaligus (*full load*) atau interval terjadwal (*incremental load*). Untuk *full load*, semua data yang sudah di transformasi dimuat langsung ke *table destination*, sedangkan *incremental load* membandingkan terlebih dahulu data yang akan dimuat dengan data yang sudah ada, data *final* yang akan dimuat ke *table destination* adalah data unik yang belum ada di *table destination*.

III.2 Tools yang Digunakan

III.2.1 Visual Studio Code

Visual Studio Code (VS Code) merupakan aplikasi *cross platform* yang dapat digunakan berbagai sistem operasi seperti windows, Linux, dan Mac OS. VS Code termasuk *software* yang ringan namun kuat *editor* sumbernya dengan deskop. Menggunakan berbagai macam bahasa pemrograman seperti Java, JavaScript, Go, C++, dan masih banyak yang lainnya. Komponen dari Visual Studio juga sama seperti yang digunakan di Azure DevOps. Visual Studio memiliki lintas *platform* kode *editor* yang ringan, dapat digunakan oleh siapa saja untuk membuat atau membangun aplikasi *web*.



Gambar 9. Logo Visual Studio Code

Sumber

:

https://commons.wikimedia.org/wiki/File:Visual_Studio_Code_1.35_icon.svg

Pada kerja praktik ini, VS Code digunakan sebagai teks editor untuk melakukan proses otomatisasi ETL menggunakan Bahasa Pemrograman Python. VS Code pun memiliki banyak sekali *extensions*, namun untuk pengembangan proyek ini hanya memerlukan *extensions* Python. *Extensions* Python digunakan agar VS Code dapat menjalankan Bahasa Pemrograman Python.

III.2.2 Telegram

Telegram adalah sebuah aplikasi layanan pengirim pesan instan *multiplatform* berbasis awan yang bersifat gratis dan nirlaba. Klien

Telegram tersedia untuk perangkat telepon seluler dan sistem perangkat komputer. Para pengguna dapat mengirim pesan dan bertukar foto, video, stiker, audio, dan tipe berkas lainnya.



Gambar 10. Logo Telegram

Sumber :

https://id.m.wikipedia.org/wiki/Berkas:Telegram_2019_Logo.svg

Para Kerja Praktek ini, Telegram digunakan sebagai media komunikasi dengan pembimbing industri.

III.2.1 Task Scheduler

Task Scheduler adalah sebuah komponen pada Windows yang menyediakan kemampuan untuk mengatur jadwal aktivitas seperti *pram* atau *script* pada rentang waktu yang telah ditentukan.



Gambar 11. Logo task scheduler

Pada pelaksanaan KP ini, *task scheduler* digunakan untuk melakukan *run file* otomatis agar program tidak harus dijalankan secara manual.

III.2.3 Dbeaver

DBeaver adalah universal tools *database multi-platform* yang gratis untuk *developers*, *database administrator*, analis, dan semua orang yang perlu bekerja dengan *database*.



Gambar 12. Logo Dbeaver

Sumber : <https://pcsoftware.eu/gb/dbeaver-corporation/dbeaver-enterprise-edition-license-included-1-year-maintenance.html>

Pada Kerja Praktik ini, Dbeaver digunakan untuk melihat sekaligus melakukan monitoring data dari Data Core.

BAB IV

ANALISIS DAN EVALUASI

IV.1 Analisis Sistem Berjalan

Dalam penerapan proses automasi ETL ini yang menjadi *source* utama adalah *Database Core*. Pertama data dari *database* ini diambil sesuai kebutuhan yang diinginkan klien. Semisal klien membutuhkan data total jumlah SMS yang terkirim di Jawa Barat setiap harinya, maka yang menjadi *source* untuk proses ini adalah *table SMS* pada *Database Core*.

Pertama data pada *table SMS* diambil lalu dipindahkan ke *dataframe*, lalu dilakukan pemilihan kolom yang penting untuk diketahui oleh klien. Karena tidak semua kolom pada *table SMS* yang penting untuk diketahui oleh klien. Setelah kolom yang diperlukan dipilih, perlu diberikan keterangan kolom tambahan secara statis untuk memberikan keterangan kepada klien.

Setelah data selesai diolah sesuai dengan kebutuhan klien, data tersebut dipindahkan ke *destination table* untuk diberikan kepada klien. Pemindahan ini bertujuan untuk memudahkan klien dalam melakukan monitoring terhadap jumlah SMS yang terkirim di Jawa Barat.

Setelah program untuk memindahkan data berhasil dibuat, diperlukan file *.bat* untuk dapat menjalankan program tersebut yang nantinya dapat diatur di *task scheduler* untuk menjalankan program tersebut secara otomatis setiap harinya. Karena jika hanya membuat program untuk memindahkan file saja, diperlukan *user* untuk menjalankan file tersebut agar data dapat dipindahkan. Maka dari itu proses automasi *run file* sangat penting dalam proses pemindahan data ini.

IV.2 Evaluasi Hasil Analisis

Dari hasil analisis yang dilakukan dapat dilihat bahwa proses ETL hanya memerlukan waktu yang sangat singkat untuk memindahkan data dari *source table* ke *destination table*. Pemindahan 10000 data secara manual mungkin memerlukan waktu satu hingga dua jam, namun dengan diterapkannya proses ETL proses pemindahan data tersebut hanya dalam kurun waktu kurang dari sepuluh menit.

IV.3 Kesimpulan Kebutuhan Aplikasi yang Dikembangkan

Dari hasil analisis dan evaluasi yang dilakukan dapat disimpulkan bahwa proses pengelolaan data menggunakan ETL ini memang penting dilakukan. Karena dengan diterapkannya ETL ini dapat memangkas waktu kerja untuk memindahkan data sehingga alokasi waktu untuk pemindahan data ini tidak banyak dan dapat produktif untuk melakukan pekerjaan yang lainnya.

BAB V

PERANCANGAN

Perancangan yang dilakukan pada automasi ETL Data Core ini adalah sebagai berikut:

V.1 Menentukan *Source Table*

Source dari setiap data yang akan dilaporkan kepada klien harus ditentukan terlebih dahulu. Seperti laporan jumlah SMS di Jawa Barat maka yang menjadi *source* adalah *table SMS*, laporan jumlah *payload* di Jawa Barat maka yang menjadi *source* adalah *table payload*.

V.2 Menentukan Pengurangan dan Penambahan Kolom

Setelah *source table* ditentukan, diperlukan pengurangan ataupun penambahan kolom terhadap data yang didapat. Karena tidak semua kolom penting untuk diketahui oleh klien, dan jika diperlukan kolom untuk keterangan yang perlu diketahui oleh klien untuk memudahkan klien dalam melihat laporan yang didapat.

V.1 Menentukan Jadwal Pengambilan Data

Dalam tahap perancangan yang dilakukan terakhir adalah menentukan jadwal untuk pengambilan data. Penjadwalan ini terbagi jadi 4 yaitu harian, mingguan, bulanan, dan tahunan. Jika data yang akan dilaporkan untuk harian, maka penjadwalan *run file* dilakukan setiap hari, jika data yang akan dilaporkan untuk mingguan, maka penjadwalan *run file* dilakukan setiap seminggu sekali.

BAB VI

IMPLEMENTASI

Data Core merupakan sebuah data yang menampung data terkait jumlah *subscriber* di suatu area, jumlah *payload* di suatu area, jumlah pengiriman SMS di suatu area, dll. Sehingga dibuatkan proses automasi ini adalah untuk memudahkan pelaporan terhadap *client* setiap harinya terkait data data tersebut. Namun, untuk untuk menampilkan real data dari Data Core tidak diizinkan. Maka dari itu, penulis hanya bisa menampilkan proses pembuatan automasi ETL menggunakan data *dummy* dari internet. Struktur data ini terdiri dari angka-angka saja yang berisikan tentang pembakaran kalori.

VI.1 Proses Pengambilan data

Yang harus dilakukan untuk mengambil data ini adalah mengambil langsung data dari *Database* Core, untuk cara pengambilannya seperti berikut.

```

6  conn = psycopg2.connect(database="database_name",
7  |         |         |         |         |         |         |         |         |
8  |         |         |         |         |         |         |         |         |
9  |         |         |         |         |         |         |         |         |
10 |         |         |         |         |         |         |         |         |
11 conn.autocommit = True
12 cursor = conn.cursor()

```

Gambar 13. Codingan 1

Seperti yang bisa dilihat di atas bahwa pada *line* 6 dilakukan koneksi ke *database* sesuai dengan *name*, *user*, *password*, *host* dan *port* yang akan dituju pada *database*. Dan di *line* 11 dan 12 dilakukan konfigurasi untuk koneksi *database* tersebut. Dalam gambar tersebut memang inputannya tidak sesuai dengan identitas dari *Database* Core sesungguhnya karena tidak diizinkan untuk ditampilkan.

Setelah berhasil terkoneksi dengan *database* maka sudah dapat untuk melakukan pengambilan data dari *database* tersebut, contoh pengambilan datanya seperti berikut.

```

14 sql = "COPY (select nama_kolom from nama_table ) TO STDOUT WITH CSV HEADER DELIMITER ','"
15
16 with open ("tes.csv", "w") as out_csv:
17     cursor.copy_expert(sql, out_csv)

```

Gambar 14. Codingan 2

Di *line* 14 dilakukan *select query* untuk mengambil data dari tabel yang kita inginkan, dan setelah itu tabel tersebut disimpan dalam bentuk csv seperti yang dilakukan pada *line* 16 hingga 17.

VI.2 Proses Pengelolaan Data

Setelah file csv berhasil terbentuk, maka dapat langsung dilakukan pengelolaan data yang diinginkan. Hasil data sebelum dilakukan operasi seperti berikut.

	Duration	Pulse	Maxpulse	Calories
0	60	110	130	409.1
1	60	117	145	479.0
2	60	103	135	340.0
3	45	109	175	282.4
4	45	117	148	406.0
5	60	102	127	300.0
6	60	110	136	374.0
7	45	104	134	253.3
8	30	109	133	195.1
9	60	98	124	269.0

Gambar 15. Output file 1

Lalu contohnya disini hanya akan mengambil kolom *duration* dan *calories* dan menambahkan kolom keterangan hari seperti berikut.

```
data = {
    'Duration' : df['Duration'],
    'Calories' : df['Calories'],
    'Description' : 'daily'
}
df = pd.DataFrame(data)
```

Gambar 16. Codingan 3

Dan hasil output setelah dilakukan pengurangan dan penambahan kolom menjadi seperti berikut.

	Duration	Calories	Description
0	60	409.1	daily
1	60	479.0	daily
2	60	340.0	daily
3	45	282.4	daily
4	45	406.0	daily
5	60	300.0	daily
6	60	374.0	daily
7	45	253.3	daily
8	30	195.1	daily
9	60	269.0	daily

Gambar 17. Output file 2

Dapat dilihat dari output file tersebut, yang sebelumnya terdapa 4 kolom menjadi hanya 2 kolom saja ditambah 1 kolom yang diinputkan secara statis.

VI.3 Proses *Ingest Data ke Database*

Setelah proses pengelolaan data selesai dan terbentuk *dataframe* baru, maka bisa langsung melakukan *ingeset* ke destinasi tabel yang diinginkan.

```

32 def execute_values(conn, df, table):
33
34     tuples = [tuple(x) for x in df.to_numpy()]
35
36     cols = ','.join(list(df.columns))
37     # SQL query to execute
38     query = 'INSERT INTO suropati."%s"(%s) VALUES %s' % (table, cols)
39     cursor = conn.cursor()
40     try:
41         extras.execute_values(cursor, query, tuples)
42         conn.commit()
43     except (Exception, psycopg2.DatabaseError) as error:
44         print("Error: %s" % error)
45         conn.rollback()
46         cursor.close()
47         return 1
48     print("the dataframe is inserted")
49     cursor.close()
50
51
52 conn = psycopg2.connect(
53     database="destination_name",
54     user='destination_user',
55     password='destination_password',
56     host='destination_host',
57     port='destination_port'
58 )
59
60 execute_values(conn, df, 'table_destination')

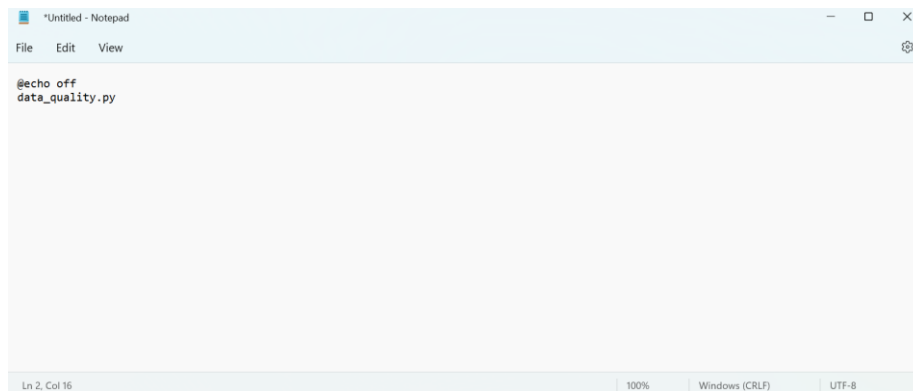
```

Gambar 18. Codingan 3

Pada *line 32* terdapat *function* `execute_values` yang merupakan *function* untuk melakukan *ingest* data dari *dataframe* yang ada ke *table destination* yang dituju. Di *line 52* terdapat koneksi ke *destination*, namun identitas yang diisikan disini bukan berupa identitas asli dari *destination core*. Dan di *line 60* terdapat pemanggilan dari *function* tersebut dengan parameter koneksi yang baru, *dataframe*, dan nama *table* yang dituju.

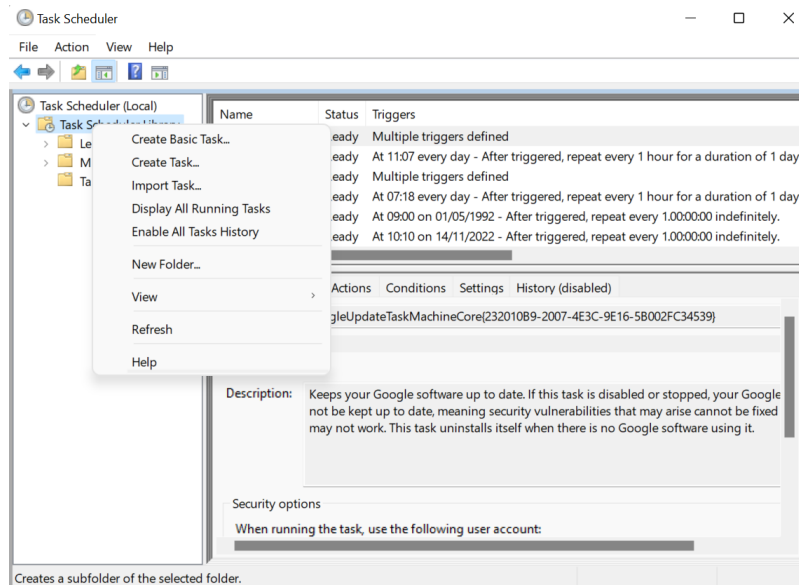
VI.4 Proses Automasi Run File

Setelah program berhasil dibuat maka diperlukan file `.bat` untuk menjalankan file tersebut secara otomatis setiap harinya. File `.bat` dibuat dengan cara membuat *notepad* dengan jenis file `.bat`. Isi dari file tersebut hanya perintah untuk menjalankan program yang sudah dibuat.



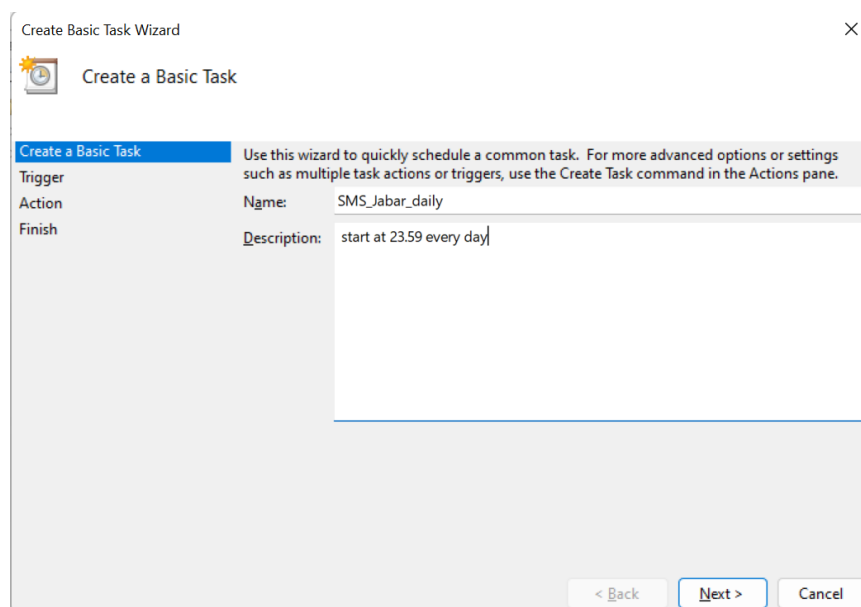
Gambar 19. File `.bat`

Setelah *file .bat* untuk menjalankan program berhasil dibuat, atur di *task scheduler* untuk menjalankan *file* tersebut sesuai ketentuan yang sudah ada di perancangan, jika harus dijalankan setiap hari maka atur untuk *file* tersebut *run* otomatis setiap harinya. Sehingga tidak perlu lagi untuk membuka *file* tersebut untuk dijalankan setiap harinya. Untuk pengaturan di *task scheduler*-nya pertama buka *task scheduler* dan buat folder baru seperti berikut .



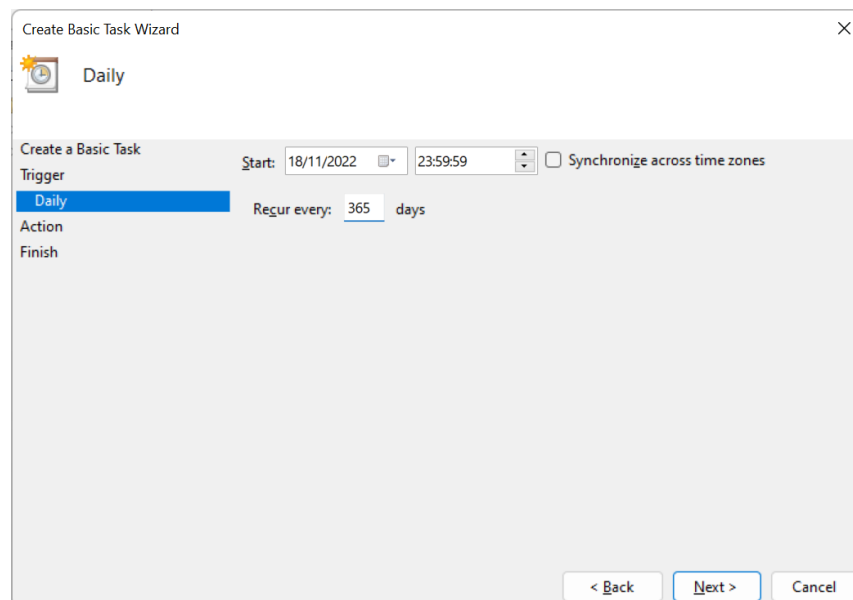
Gambar 20. Task scheduler 1

Lalu masuk ke folder yang baru dibuat dan pilih *Create Basic Task* di *action*. Lalu masukan judul dari nama *scheduler* dan deskripsinya. Contohnya seperti berikut.



Gambar 21. Task scheduler 2

Setelah itu pilih *daily* dan tetapkan untuk menentukan jam dan tanggal dimulai programnya. Contohnya seperti berikut.



Gambar 22. Task scheduler 3

Setelah itu pilih *file* yang akan dijalankan nya, dan *finish* pembuatan *task*-nya. Maka program tersebut akan *run* otomatis setiap harinya.

BAB VII

PENUTUP

VII.1 Kesimpulan

Dari hasil penjelasan di atas dapat disimpulkan bahwa pemindahan data dari *source* ke *destination* sudah bisa secara otomatis. Dengan diterapkannya automasi ETL ini waktu yang digunakan untuk memindahkan data menjadi sangat singkat dan meningkatkan produktifitas karena alokasi waktu untuk mengerjakan pekerjaan lain menjadi lebih banyak.

VII.2 Saran

Pada pengelolaan data kali ini masih belum optimal, karena data yang sudah diambil dari *source table* belum dilakukan tindakan apapun. Artinya jika terus seperti ini data dari *source table* akan menumpuk dan memenuhi *storage database*. Diharapkan untuk kedepannya dibuatkan proses untuk penghapusan data dari *source table* jika memang data tersebut sudah tidak dapat digunakan lagi agar terhindar dari penumpukan data yang tidak berguna.

DAFTAR PUSTAKA

- Wijaya, Rahmadi dan Pudjoatmadjo, Bambang. “Penerapan Extraction-Transformation-Loading (ETL) Dalam Data Warehouse (Studi Kasus : Departemen Pertanian)”. Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI) 05, no.2, (2016): 75.
- Faradilla A. (2022, September). Apa Itu Python? Yuk Kenalan dengan Bahasa Pemrograman Ini. <https://www.hostinger.co.id/tutorial/python-adalah>.
- Tasari, Gusmi. (2021, April). Mengenal Visual Studio Code. <https://www.gamelab.id/news/468-mengena-visual-studio-code>.
- Amaluidin, Muhammad. (2021, Januari). Aplikasi Telegram, Beserta Keamanan Penggunaanya. <https://www.jabarnews.com/sekolah-juara/aplikasi-telegram-beserta-keamanan-bagi-penggunaanya/>.
- Belang, Semut. Sejarah Google Chrome Lengkap dari Awal Terciptanya. <https://semutaspal.com/sejarah-google-chrome/>.
- Ramadhan, Ashari. (2022, Juni). Tutorial Install DBeaver pada Windows, SQL Client Ramah Pemula. <https://dqlab.id/tutorial-install-dbeaver-pada-windows-sql-client-ramah-pemula>.